

# 質問応答のための質問文と知識文の間の意味ベースでの精密な照合方式

竹原 一彰<sup>\*</sup> 安部 建助<sup>\*\*</sup> 安田 智成<sup>\*\*</sup> 韓 東力<sup>\*\*\*</sup> 原田 実<sup>\*\*\*</sup>

青山学院大学 理工学部 機械創造工学科<sup>\*</sup> 青山学院大学 理工学部 情報テクノロジー学科<sup>\*\*</sup>,<sup>\*\*\*</sup>

## 1. はじめに

日本語をベースとした質問応答の研究では、村田らが質問文と検索されたパッセージの係り受け木での対応語間に IDF や EDR の語彙による類似度を用いて、最も類似したパッセージから解を得ている [1]。この研究では意味解析を行っていないので語意の精度は非常に低く、また語間の関係も係り受け関係を用いているので深層の意味に基づく重要度を把握できず、解答精度は 50% から 70% 程度である。他の研究も同様で精度に問題がある。そこで、日本語文章の意味解析に基づき高精度の質問応答システムを構築するための基礎研究を行う。文章内容の形式表現方法を調査の結果、質問文と知識文（検索文）の精密な照合を行うために、Sowa の提案する概念グラフの理論が最適だと判断した。質問文と知識文に対する概念グラフの類似度を計算することによって、これまでより文章間の精密な照合を行うことができるようになる。

## 2. 自然言語の概念グラフ表現

概念グラフとは全ての事象を概念とその関係（リレーション）で表現したものである。自然言語で書かれた知識は概念グラフで形式表現できる。ノードが概念に対応し、その概念のインスタンスや多重度はリファレントと呼ばれ、概念とともにノード情報として保持される。リレーションは名前つきアークとして表現される。さらに様相（時制・否定）もノードの持つ属性として表現可能である。本照合システムでは図 1 に示すような形式で概念グラフを表現している。

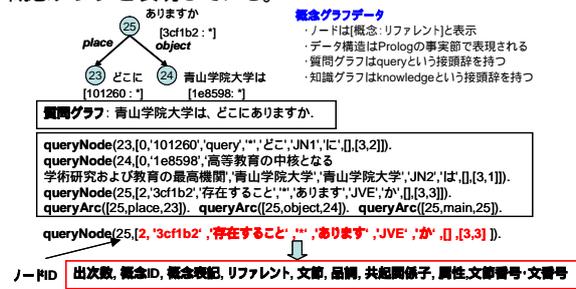


図 1 概念グラフのデータ構造（質問グラフ）

## 3. グラフの照合定理

照合の正しさは、「あるグラフ G が真であるためには、真であることが確定しているグラフ T (または T の部分グラフ) への特殊化の系列が存在することである」 [2] という定理に基づく。

A precise matching based on semantic analysis between the question sentence and knowledge sentence for question-answering Kazuaki Takehara\*, Kensuke Abe\*\*, Tomonari Yasuda\*\*, Dongli Han\*\*\* and Minoru Harada\*\*\*  
 \*Department of Mechanical Engineering, Aoyama Gakuin University.  
 \*\*Department of Integrated Information Technology, Aoyama Gakuin University.  
 \*\*\*Department of Integrated Information Technology, Faculty of Science and Engineering, Aoyama Gakuin University.

しかし、自然言語のような多様性のある知識をベースとしたグラフの照合にはこの定理は制約が厳しすぎる。そこで、本照合システムではグラフの類似度に閾値を設け、それを上回れば定理の条件を満足するように制約をゆるめて利用している。グラフ類似度(スコア)は下式に示すノード類似度とリレーション類似度(それぞれ最高値 100)の和である。完全にグラフが一致するときはスコア 200 となる。

$$\text{ノード類似度} = \frac{\sum \text{照合ノードペアの類似度}}{\text{質問グラフのノード数}} \times 100$$

$$\text{リレーション類似度} = \frac{\sum \text{照合ノードペア間のリレーション数}}{\text{質問グラフのアーク数}} \times 100$$

## 4. 照合システム

本システムは、1つの質問グラフとそれに解答を含んでいような知識グラフの集合を入力とし、照合結果として、ノードの照合関係、解答とそのスコアをファイルに書き出す。

### 4.1. 照合のアルゴリズム

質問グラフを張る木の根 qst からスタートして、質問グラフのノード qn を縦型に訪問しながら知識グラフのノード kn と照合していく。また照合を行いながらグラフ類似度を計算する。アルゴリズムの概要を以下に示す。また各種関数の意味を表 1 に示す。

表 1 各種関数とその意味

| 関数             | 意味                       |
|----------------|--------------------------|
| children(n)    | nの未照合の子ノード順序集合を返す        |
| descendants(n) | nの未照合の子孫の順序集合を返す         |
| outdeg(n)      | nの出次数を返す                 |
| bind(qn, kn)   | qnとknを照合ペアとする(このペアは類似度0) |
| rel(n1, n2)    | n1からn2へのリレーションを返す        |
| inrel(n)       | nへの入力辺のラベルを返す            |
| getElm(S)      | 引数の順序集合の次の要素を返す。なくなると偽。  |
| fill(qn, kn)   | ノード対qnとknが表2の条件を満たすならば真  |

表 2 ノードが照合する条件

|        | 質問ノード            | 知識ノード          |
|--------|------------------|----------------|
| 概念類似度  | 0.27以上(経験的に定めた値) |                |
| リファレント | 一般概念(*)          | 何でも良い          |
|        | 固有概念             | 質問と同じリファレントを持つ |
| 属性     | 同じ属性をもつ          |                |

$$\text{概念類似度} = \max \left( \frac{2 \times d_c}{d_q + d_k} \right) \begin{matrix} d_q, d_k: \text{それぞれの概念の概念深さ} \\ d_c: \text{dq, dkの共通概念の概念深さ} \end{matrix}$$

```
main(){
    while(kst getElm({kn | outdeg(qst) outdeg(kn)
fill(qst, kn)})){
        bind(qst, kst);
        matching(qst, kst);
        if(リレーション類似度 閾値)
            照合結果をファイルに書き出す;
    }
}
```

```

matching(qn, kn){
  while(nqn getElm(children(qn)){
    while(dkn getElm(descendants (kn))){
      if(rel(nqn,nqn)=inrel(dkn) fill(nqn,dkn)){
        bind(nqn,dkn);
        matching(nqn,dkn);
        break;
      }
    }
  }
  if(dkn が偽){
    bind(nqn, );
    if(nqn が葉でない){
      while(dqn getElm(children(nqn))){
        rel(qn,nqn)を rel(qn,dqn)につなぎかえる;
      }
    }
    matching(qn,kn);
  }
}
}

```

具体的にこのアルゴリズムを図 3 に示すようなグラフ(この例では同じアルファベットのノード対のみ概念類似度が 0.27 以上になるとする)に対して適用すると、点線のようにノード対が照合する。照合した部分を共通グラフと呼ぶ。重要なところとしてノード Q(3)とノード K(6)の照合されるときリファレント\*は c にインスタンス化(特殊化)される。またノード Q(4)とノード K(5)は同じ接続関係(object)かつ同じ概念を持つがリファレントが異なるので照合されない。

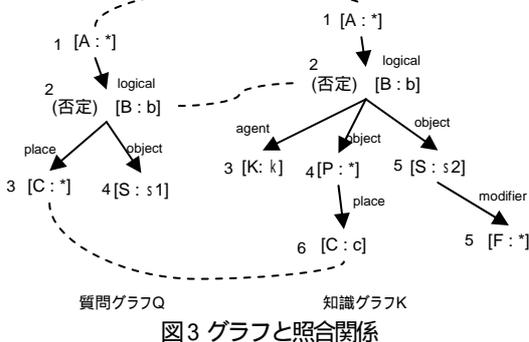


図3 グラフと照合関係

### 5. Type Expansion

言語表現の多様性の差を吸収するために概念グラフの Type Expansion を行う。グラフ操作としては、深層関係を考慮しながら、ある概念を表現するノードを意味的に同値なグラフで置換する。簡単な例として、「メールを送る」を「メールを送信する」、「e メールを送る」などの同値な置き換えが可能である。このルールは現在人手で記述されているが国語辞典などから Type Expansion ルールの自動取得も視野にいれ、表 3 に示すフォーマットで、Type Expansion データベースに格納する。

表3 Type Expansionルールの構成要素

|         |                                           |
|---------|-------------------------------------------|
| 条件グラフ   | このグラフが部分グラフであるときType Expansionを適用できると判定する |
| 定義グラフ   | 条件グラフに置き換わる同値なグラフ                         |
| コンストラクタ | 置き換える際の、条件(接続関係など)を記述しておく                 |

質問グラフ Q に対して rules(G)を「グラフ G に適用できる Type Expansion のルール数」と定義すれば、このとき Q の同値グラフ

は、 $2^{\text{rules}(Q)}$ 個ある。同様に知識グラフの集合  $K$ ( $j$  個の知識グラフ  $K_1 \sim K_j$  からなる)の同値グラフの数は  $\sum_{i=1}^j 2^{\text{rules}(K_i)}$  個あるので、照合は  $2^{\text{rules}(Q)} \times \sum_{i=1}^j 2^{\text{rules}(K_i)}$  通りのグラフの組み合わせについて行う(図 4)。

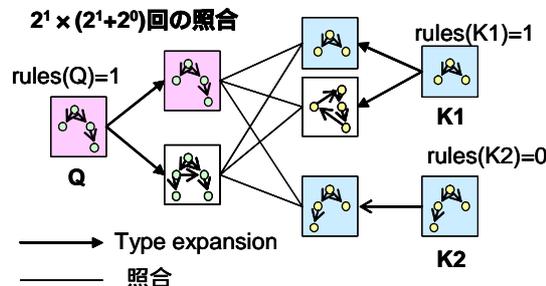


図4 照合のパターン

### 6. 解の抽出

図 4 のように照合が行われ、照合結果は回答とそのスコアという形で保存される。抽出された複数の解のそれぞれについてスコアの総和を計算する。その中で、スコアの高いものから順位つきで解を抽出する。

### 7. 精密化

Q: 「ナイル川はどこを流れていますか?」という質問に対して K: 「ミネソタ州の南北をミシシッピ川が流れている。」という知識文が検索されてきたとする(図 5)。これまでの方法ではグラフ Q とグラフ K の構文木の形、概念(概念)の並びが等しいので、[どこを]=[ミネソタ州の南北]と解が抽出されてしまうのであるが、本照合方式を用いれば、概念 ID が 44495e(名称で捉えた河川)で示されるノードのリファレントが表 2 の条件(3. グラフの照合定理)に反するので、このノード対は照合されない。その結果 Q と K の照合は類似度が閾値を超えなくなるので間違っただけの解を抽出しない。属性の導入も同様な効果がある。

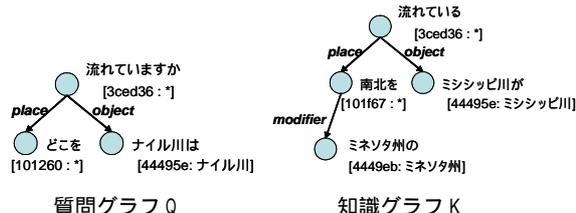


図5 質問グラフと知識グラフ

また、Type Expansion により、今までは同じ意味を表すグラフであっても語意の類似度や構文木の違いにより、照合がうまくいかなかったグラフに対しても、一貫した方法で照合できるようになる。

### 参考文献

[1] 村田真樹、内山将夫、伊佐原均: “類似度に基づく推論を用いた質問応答システム”  
 [2] John F. Sowa: “Conceptual Structures: Information Processing in Mind and Machine”, 1984